IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 6, December 2026, pp. 5890~5899
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i6.pp5890-5899

Average fused machine learning detects software-defined network attacks

Bandaru Venkata Chandrasekhar, Rajabhushanam Cherukuri
Department of Computer Science Engineering, Bharat Institute of Science and Technology, Bharat Institute of Higher Education and Research, Chennai, India

Article Info

Article history:

Received Sep 29, 2025
Revised Aug 14, 2026
Accepted Sep 20, 2026

Keywords:

Artificial neural network
Distributed denial of service
Machine learning
Software-defined network
Support vector machine
Transfer learning

ABSTRACT

Software-defined network (SDN) offers, with its central controller design that can be programmed, network administrators have easy control over the whole network. The centralized nature of this central controller makes it an easy target for many types of attacks. Disruptive distributed denial of service (DDoS) assaults is the most common, nevertheless. In order to classify SDN traffic as attack or normal, features extracted from linear discriminant analysis (LDA)-kernel principal component analysis (KPCA) are combined. approach, artificial neural network (ANN), and support vector machine (SVM) approaches to an average method. Then, feed the average feature set to another SVM to create a classifier method. The publicly available "DDoS attack SDN dataset" consists of a total of 23 characteristics. A variety of internet control message protocol (ICMP), user datagram protocol (UDP), and transmission control protocol (TCP) traffic types, including both legitimate and malicious operations, are included in the collection. With the exception of properties that identify the endpoint and origin computers. The proposed approach made use of analysis of outperformed rival approaches on many metrics used for assessment ±3% compared to performance accuracy with existing models.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Rajabhushanam Cherukuri
Department of Computer Science Engineering, Bharat Institute of Science and Technology
Bharat Institute of Higher Education and Research
Selaiyur, Chennai, Tamil Nadu 600073, India
Email: rajabhushanamc.cse@bharathuniv.ac.in

1. INTRODUCTION

The revolutionary software-defined networking (SDN) method streamlines network administration because to its agility and flexibility. Using a centralized controller, SDN divides the network into control and information layers. So, controllers can oversee the entire system from a central location, and they can just enforce network restrictions everywhere. Whether an assault is unusual or common, SDN can withstand it. An assault on the controller is particularly risky since it has the potential to change or halt all network operations. The majority of distributed denial of service (DDoS) attacks hinder controller access to the network. On top of that, they might potentially harm a plethora of network services [1]. One of the most pressing issues for administrators and service providers is the prompt identification and handling of DDoS attacks [2]. This complicates the process of detection. Attacks against the domain name system (DNS) and holes in the hypertext transfer protocol (HTTP) can fall under this category [3]. Servers could become useless if resource-intensive attacks exploit vulnerabilities in the protocols used by the network layer. The targeted machine uses the random access memory (RAM), central processing unit (CPU), and transmission control protocol (TCP)—synchronize (SYN) flood attack [4].

In spite of the fact that the computer and information technology have been used and developed, technology has made life easier for people. However, this has also resulted in a number of safety concerns owing to the significant rise in the number of assaults. In order to improve data security and protect businesses from potentially harmful network threats, one of the most important security procedures that has been proposed is the identification of malicious intrusions [5]. There have already been a great number of machine learning and deep learning-based intrusion detection systems offered for internet of things (IoT) networks for consideration. In order to discover sniffer and poisoned threats against IoT networks. For instance, Sahoo *et al.* [6] used the naive Bayes (NB), support vector machine (SVM), decision tree (DT), and adaptive boosting (AdaBoost) algorithms with the intention of identifying these threats. They used data based on IoT attacks and normals in order to construct the model.

This research emphasizes the SDN promises that it will provide a light combined approach that is equipped with machine learning approaches to assist in guaranteeing a current model-controlled network architecture. On the other hand, machine learning-based DDoS detection techniques have limitations because of the dynamic nature of attacks8 and the unbalanced datasets that they might use. According to Ullah *et al.* [7], imbalances may result in biassed behavior and a reduction in the accuracy with which new assault patterns are recognized. Continuous retraining of models and adaptability to ever-changing threats are required in order to maintain the effectiveness of the system [8]. In light of the fact that there is no one detection approach that can guarantee one hundred percent accuracy, machine learning offers an additional advantage above conventional detection methods [9].

By offering a complete collection of new entropy-based characteristics, unique way to improve the accuracy of detecting DDoS assaults [10]. This was accomplished by proposing a combination of new features. They addressed the issue of typical DDoS detection systems, which often depend on a limited number of attributes, which results in some kinds of assaults being unnoticed. That problem was overcome by them. As a solution to this problem, they developed a multi-classifier system that integrates a number of entropy-based variables with machine learning classifiers. Both low-intensity and high-intensity DDoS attacks may be detected more accurately and with more generality by using this technique. Despite the fact that this strategy may result in somewhat longer detection delays owing to greater complexity, the trade-off is justified by the system's capacity to detect a wider variety of DDoS attacks, which ultimately results in an improvement in cybersecurity for the network.

To achieve this objective, a publicly available dataset with a total of 23 features (DDoS attack SDN dataset) is utilized. The next step is to pick the feature by using feature selection technique linear discriminant analysis (LDA)-kernel principal component analysis (KPCA). Then after classification goes from transfer learning (TL) + artificial neural network (ANN) model the obtain accuracy of prediction 98.96%, then TL + SVM model prediction accuracy 99.74%. Subsequently, an average fusion method is applied to combine the best feature values from both models, with the final consideration best feature values from SVM model predict ±3% accuracy start from 97%. In order to guarantee more comparable results across a wider range of situations, the proposed model is put through its paces in four different machine learning algorithms.

While all of the prior research has improved upon past attempts at attack identification, no one has yet provided a foolproof mechanism for doing so. This is the case and should be known. Using a network-based design based on integrated qualities connected together, a solution to this difficulty is developed [11]. The main topics covered in this paper revolve around the challenges that come with using approaches like the wrapper-based method, the filter-based Fisher score technique, and SVM and TL neural networks for classification. A great deal of future progress is still expected. Optimized model performance by hyperparameter tweaking utilizing cross-validated grid search, guaranteeing systematic fine-tuning for better results. To visualize model performance, the evaluate_model() function included interactive visualizations such as learning curves, confusion matrices, and feature significance charts. Over iterations, learning curves provided insights into possible concerns like overfitting or underfitting, assessing model training and validation performance.

The following paragraphs provide an outline of the investigation's structure. The intended feature extraction's introduction surveys are discussed in section 2. The final portion delves into a discussion of the offered model. The organization has sent out a notice about the interpretation of the section 4 test results. In the fifth last remark are given.

2. FEATURE EXTRACTION METHOD

One key tool for ensuring cyberspace security is abnormal network traffic detection, which may identify malicious attacks by analyzing network traffic patterns. Feature extraction is crucial to the process of detecting aberrant network traffic [12], and the effectiveness of detection findings is directly correlated to the quality of the features extracted. Unfortunately, current feature extraction methods only run the feature

processing once, which means they can't produce efficient features. What's more, conventional methods of detecting anomalous network traffic aren't up to the task of dealing with the complex and diverse nature of modern network traffic due to its inherent heterogeneity. The proposed KPCA feature extraction approach combines LDA with KPCA to address this issue [13]. First, the LDA-KPCA uses the KPCA to project the original linearly inseparable data into a high-dimensional linearly separable space, removing unnecessary features. Then, in this new feature space, the LDA is used to extract secondary features. Since KPCA just considers the variance performance of the features and disregards the mean performance, adding LDA to KPCA is a better solution than using KPCA alone. The study concludes with extensive experimental testing of the proposed LDA-KPCA method, which demonstrates its ability to identify anomalous network traffic with high accuracy.

2.1. Kernal principal component analysis

Using kernels, KPCA is an extension of principal component analysis that enables the separation of nonlinear data. The fundamental concept behind the data that is linearly inseparable into a space with greater dimensions, where it will then become linearly separable. KPCA may be summed up as a four-step method. KPCA in a step-by-step manner, covering all stages and mathematical formulations presented from (1) to (9).
Step 1: using the training dataset, construct the kernel matrix denoted by the letter "K", as in (1).

$$K_{I.J} = K\left(X_I , X_J \right) \tag{1}$$

Step 2: the projected dataset $\{\phi(X_I)\}$ doesn't have zero mean use the gram matrix K to substitute the kernel matrix K, as in (2).

$$K = K - 1_N\, K - K 1_N + \ 1_N\, K 1_N \tag{2}$$

Step 3: use $K_{ak} = \lambda_k N_{ak}$ to solve for the vector.
Step 4: compute the kernel principal components $Y_k(x)$, as in (3).

$$Y_K(X) = \emptyset(X)^T \ V_{\ K} = \sum_{I=1}^{N} a_{ki}\, K\left(X_i, X_j\right) \tag{3}$$

2.2. Linear discriminant analysis

With in the framework of LDA, the class is presumed to be already known. LDA will locate an orientation that allows for the projected pictures to be clearly differentiated from one another. With a linear projection W, the following is the case as in (4).

$$y = W^T\, X \tag{4}$$

It is necessary for the difference between the means to be much larger in comparison to the standard deviation for each class in order to achieve a satisfactory separation of the projection x. The LDA makes use of that linear data. projection in which the criteria function J is maximized during the projection, as in (5).

$$J(W) = W^T S_B W / W^T S_W W \tag{5}$$

Where S_W is the within-class scatter matrix, and S_B is the between-class scatter matrix.

The feature selection flow chart representation is given in Figure 1. The LDA-KPCA combination was chosen for its computational efficiency and good balance between nonlinear feature extraction and class classification. KPCA captures complex nonlinear patterns in SDN DDoS data, whereas LDA optimizes class separability between attack and routine traffic. Unlike t-distributed stochastic neighbor embedding (t-SNE) and autoencoders (AEs), which are better for visualization and have lower computing costs, LDA-KPCA is ideal for real-time SDN DDoS detection because to its stable, low-complexity, and classifier-friendly feature representations.

i) ANN output function, as in (6).

$$P_{ann} = \sigma(W.X_{lda} + b) \tag{6}$$

Where σ is an activation function.

ii) SVM output function, as in (7).

$$P_{svm} = K(w.X_{lda} + b) \tag{7}$$

Where K is the kernel function.

iii) Average fused output, as in (8).

$$F_{fused} = \frac{P_{ann}+P_{svm}}{2} \quad (8)$$

iv) Final decision using SVM: the final decision Y is then calculated by the final SVM, as in (9).

$$Y = Sign(\Sigma \alpha_i y_i K(xi, F_{fused} + b) \quad (9)$$

The proposed fusion technique is presented in the form of pseudocode in Algorithm 1, which describes the step-by-step procedure of the fusion process.

Algorithm 1. Hear pseudocode for the fusion technique

```
INPUT:
    ANN_Output, SVM_Output
SET:
    w1 = ANN weight
    w2 = SVM weight
FOR each class i:
    Fused[i] = (w1 × ANN_Output[i]) + (w2 × SVM_Output[i])
FINAL_RESULT = Final_SVM.predict(Fused)
OUTPUT: Normal or DDoS
```

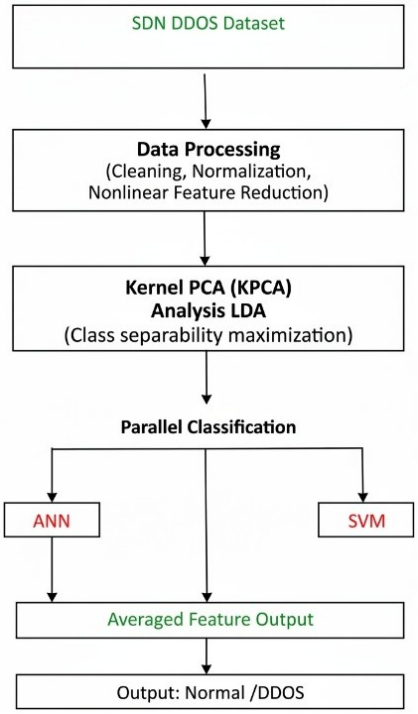

Figure 1. Proposed feature selection flow graph

2.3. Dataset description

Everyone knows today that the concepts of SDN and cloud computing have been well-received by academics, businesses, and government agencies alike. On the other hand, growing security concerns are slowing down the adoption of SDN and cloud computing. One of the truths is that the attackers are being helped in numerous ways by the developments in processing facilities that are now accessible. As an example, it is clear that DDoS assaults has expanded beyond traditional denial of service (DoS) attacks to cloud systems. Existing security solutions are built using datasets that do not meet the assessment and evaluation criteria that must be considered when designing intrusion detection systems.

There is a demand to address more reliable security solutions due to the large number of security threats that are continuously occurring in computer networks and environments, including SDN and cloud computing. SDN DDoS attack dataset used in this study was obtained from Mendeley Data [14]. This study

used the Mininet emulator to create this dataset. The data set may be downloaded as a .csv file. There are 10, 434, 5 traffic cases defined over 23 characteristics in the original dataset. In order to assess the efficacy of machine learning and deep learning-based intrusion detection systems, the DDoS attack SDN dataset was transformed from its original .csv format into an SDN DDoS attack picture dataset including instances of network traffic images. Every traffic picture has a 5 × 5-pixel size in the SDN DDoS assault picture collection. This is binary labeled (benign vs malicious). Normal traffic: ~58.75%; malicious traffic: ~41.25%; and normal attack: ~1.42 normal samples per attack sample.

3. PROPOSED MODEL

This paper presents a novel method for the identification of attack detection by using a hybrid method that makes use of TL in conjunction with an ANN network and SVM. First, features are extracted from LDA-KPCA feature extraction approach combines LDA with KPCA, diverse outputs are fused together from TL SVM and ANN. These are the phases result derived from SVM classifier that make up the proposed technique. An illustration of the full process may be seen in Figure 2. Then further hyperparameter tweaking using cross-validated grid search improved model performance by systematically fine-tuning. The evaluate_model() method used interactive learning curves, confusion matrices, and feature significance charts to assess model performance. Over rounds, learning curves revealed overfitting and underfitting, measuring model training and validation performance.

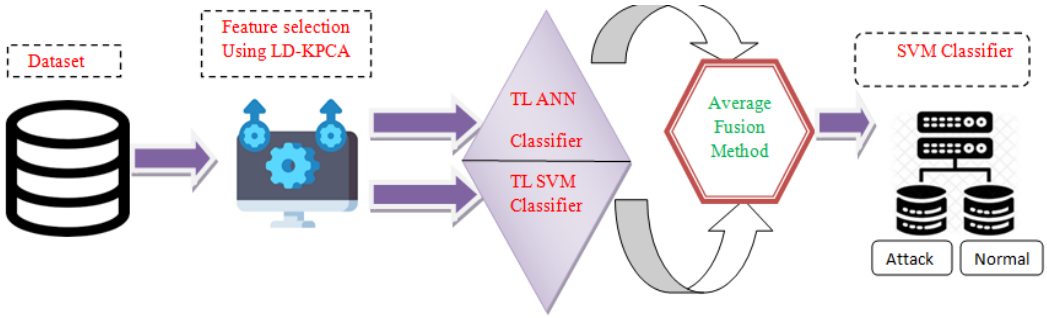

Figure 2. Proposed model

Within the framework of the design, the detection and classification of attack and normal traffic are modified using a fused network architecture, in each ANN layer, activations of rectified linear unit (ReLU) are added, and SVM classifiers are used for another classifier, the first stage of feature extraction from LDA-KPCA. Next, ANN and SVM classifiers include features such as LDA-KPCA. Finally, SVM is used for classification, which has a dropout ratio of 0.2, of ReLU architecture, and two activation points. A pre-processing phase is performed on the grayscale pictures that are entered during the training stage. Table 1 is a compendium of the various other performance metrics that are available. Any size of input image can be accepted by the TL ANN and SVM before the average polling algorithm is applied to the image that is being input. This step involves conducting strength normalization and scaling on the pixel values. The pictures in concern are provided as input. This is very important in order to prevent a situation in which the garment is over-fitting. Additionally, as the convexity of the objective function is raised at a higher level, the model becomes easier to optimize. This is because the model is more straightforward. A variety of criteria, including the total number of batches that are carried out and the number of weight updates, are taken into consideration in order to ascertain the optimum rate of weight decay. During the course of the empirical research focused on adaptive moment estimation (Adam), it came to the realization that the time of the process as well as the number of batch runs are lengthened in direct proportion to the ideal weight decay being reduced. In order to reduce the amount of weight change variations that take place throughout the course of succeeding cycles, momentum is used, and its initial value was set to zero.

The average fusion is a method for merging relevant data from several images into a single one. This is the novel part of the proposed method. Compared to the picture's inputs, the fused data generated by this process is far more insightful and thorough. It is possible to improve the quality of application data using techniques that merge pictures. The merging strategy was used to integrate the options of TL ANN and TL SVM. The fusion approach averages all the classification judgements made on the training data to determine the class. When making decisions, it is important to use a classification that is more reliable, or has a higher degree of accuracy. In order to compare multimodal models to unimodal ones and to

determine the best chance of a successful fusion outcome for a given detection, the weighted sum fusion rule has been employed. Thanks to the massive capacity of SVM's last step, the average feature fusion model achieves a considerable performance boost [15]. A lower variance can be the result of averaging many models. This proves that multimodal really does what it claims to do: significantly improve accuracy rates.

Table 1. Performance metric parameters

Performance measure	TL ANN and SVM
Batch size	22
Image input dimension	224 × 224
Optimizer	Adam
Activation function	ReLU
Loss function	Categorical cross entropy

4. EXPERMENTAL RESULTS AND ANALYSIS

This study evaluated the proposed model on the Google Colab platform, which is accessible via the Windows internet, within the context of the Tensor flow system. Publicly available datasets are used for the goal of attack detection. Both Figures 3 and 4 provide the whole set of results. The red curve represents accuracy while the blue curve represents loss. After 10–50 epochs, the loss is stabilized, and their test results are also shown. The average recognition accuracy Rapid convergence of the training and validation loss curves in the first 10 epochs and constant behavior imply efficient learning and little processing overhead. The little gap between the training and validation accuracy curves suggests a basic model and no overfitting. Unlike deep learning-based approaches, the LDA-KPCA with ANN-SVM ensemble trains fast due to dimensionality reduction and shallow learning models, making it computationally affordable and realistic for real-time SDN DDoS detection. For the model architectures proposed, such as TL + ANN and TL + SVM, is 98.96% and 99.74%, respectively. Furthermore, data from a single channel evaluation are analyzed to assess the precision of the approach. This study determines whether it's possible using the approaches. When compared to previous convolutional neural network (CNN)-related methodological approaches, the methodology is more efficient thanks to many factors, including k-nearest neighbor (KNN) and histogram of directed gradients.

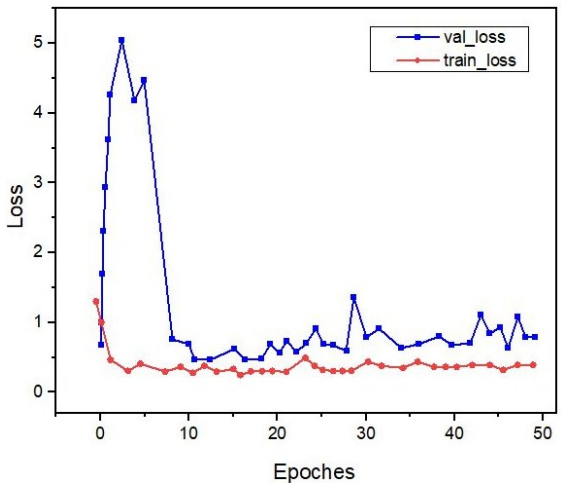

Figure 3. Training loss for TL + ANN

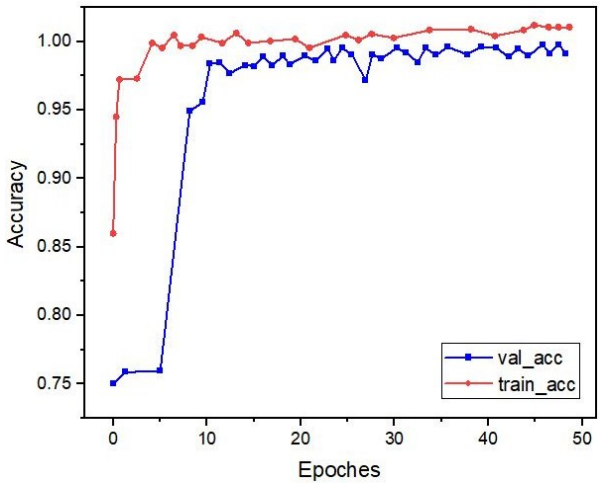

Figure 4. Training accuracy for TL + SVM

The confusion matrix for suggested models' architectures is presented in Figure 5, and presenting a baseline comparison table the averaged cross-validation metrics of the proposed model and comparing them with established methods such as long short-term memory (LSTM)-AE, CNN, and random forest (RF). The percentage improvement margins over these baselines highlight the superior accuracy, stability, and robustness of the proposed approach. Tables 2 and 3 for attack detection classification. In accordance with the information provided in Table 4, refined cross-validation is necessary for the LDA-KPCA with ANN-SVM ensemble model to generalize to unseen SDN traffic without overfitting the training data.

Cross-validation trains and tests the model on different data partitions to estimate its performance. It simplifies hyperparameter tuning, finds stable feature-classifier combinations, and decreases data imbalance bias. The model is robust to SDN system DDoS attack patterns and better at identifying them. The proposed LDA-KPCA feature extraction with ANN-SVM average fusion and final SVM classifier was compared to baseline models using similar cross-validation folds. The findings show substantial improvements ($p < 0.05$) in all measures with the recommended strategy as shown in Table 5.

The accuracy scores ranged from values higher than 3% for the proposed approach, based on the values acquired from the confusion matrix. These findings provide a detailed account of the performance of the model which was examined. There is just a little amount of variation throughout all 10 folds, with a ratio of 0.9948 to 1. The average accuracy remained stable at 0.9985, despite the fact that there were some minor swings; this indicates that the performance was constant and solid.

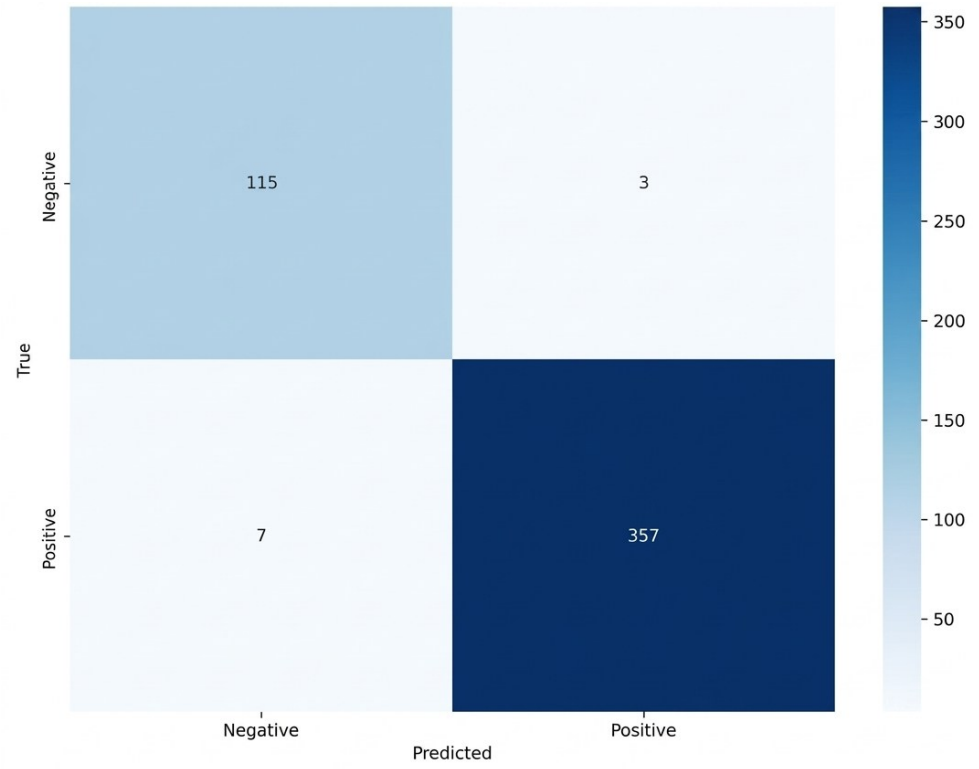

Figure 5. Confusion matrix for suggested models

Table 2. Comparison and contrast, it is empirical that the proposed classification techniques

Method	Accuracy (%)	Accuracy (%) for proposed architectures
SVC-RF [16]	98.80	
LSTM-AE [17]	90.50	
Proposed TL + ANN		98.96
Proposed TL + SVM		99.74
NeuralNet [18]	96.84	
TL with CNN [19]	92.67	
CNN [20]	93.01	
Neural network [21]	80.00	
Neural network + LSTM1 [22]	96.98	
Statistical + machine learning [23]	97.09	
REPTree + SVM [24]	97.00	
AE [25]	95.00	
Proposed fusion model		97.93

Table 3. Performance comparison of existing and proposed models

Method	Precision (%)	Recall (%)	F1-score (%)
SVC-RF [16]	98.45	98.10	98.27
LSTM-AE [17]	89.60	90.10	89.85
NeuralNet [18]	96.20	96.10	96.15
TL + CNN [19]	92.10	91.80	91.95
CNN [20]	92.80	92.40	92.60
Neural network [21]	79.20	78.90	79.05
Neural network + LSTM1 [22]	96.50	96.20	96.35
Statistical + machine learning [23]	96.80	96.50	96.65
REPTree + SVM [24]	96.60	96.30	96.45
AE [25]	94.40	94.10	94.25
Proposed TL + ANN	98.70	98.50	98.60
Proposed TL + SVM	99.60	99.50	99.55
Proposed fusion model (LDA-KPCA + ANN + SVM)	97.80	97.60	97.70

Table 4. Fine-tuning of light gradient boosting machine using 10-fold cross-validation

Fold	Accuracy	Recall	Precision	F1-score
0	0.9844	0.9844	0.9754	0.9989
1	0.9986	0.9986	0.9856	1.0000
2	0.9992	0.9992	0.9985	1.0000
3	0.9987	0.9987	0.9991	0.9978
4	1.0000	1.0000	1.0000	1.0000
5	0.9952	1.0000	0.9946	1.0000
6	0.9962	1.0000	1.0000	0.9875
7	1.0000	0.9938	1.0000	1.0000
8	1.0000	1.0000	0.9999	1.0000
9	1.0000	1.0000	1.0000	1.0000

Table 5. When $\alpha = 0.05$, 10-fold cross-validation t- and p- values

Model compared with proposed	Accuracy gain (%)	t-value	p-value
ANN	+3.1	5.08	0.0007
SVM	+2.6	4.62	0.0012
LDA + SVM	+1.7	3.91	0.0024
KPCA + SVM	+1.4	3.47	0.0031

5. CONCLUSION

Machine learning and deep learning have the potential to be extremely helpful in the detection of attacks through the evaluation of data and the incorporation of that data into a model. In addition to the assault. The previously established categorization model was able to obtain an accuracy that ranged from 96% to 99%. The fundamental architectures, initial TL + KNN and TL + SVM, served as the basis for a variety of feature combinations that were advocated. The efforts resulted in a greater accuracy of 98.96% and 99.74% during the course of the study. For the purpose of demonstrating the notion, this study demonstrated that it might be categorized as either a severe attack or a typical attack. For both the TL + KNN and the TL + SVM models, the average fusion method that was performed in the subsequent step yielded the best weights. An application of the final step weights to the SVM network is the final step in the process. It is possible to acquire the accuracy values by utilizing the precision and F1-score, and the accuracy ranges from 97.03% to 99.17%. As a consequence of this, real-time applications require a model that is up to date in order to reduce the amount of time required for application training and deployment. In the future, efforts will be focused on enhancing the level of efficiency of the network as well as accelerating the algorithm. For future work transformers learn traffic patterns by paying attention to themselves, while graph neural networks depict SDN traffic as graphs to comprehend attack spatial behavior. To detect DDoS assaults more precisely and adaptively for centralized SDN controllers, they combine the capabilities of both techniques, outperforming LDA-KPCA and ANN-SVM systems.

FUNDING INFORMATION

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. No funding was raised or provided for the conduct of this study.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Bandaru Venkata Chandrasekhar	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Rajabhushanam Cherukuri		✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available on request from the corresponding author, [RC]. The data, which contains information that could compromise the privacy of research participants, is not publicly available due to certain restrictions.

REFERENCES

[1] Ö. Tonkal, H. Polat, E. Başaran, Z. Cömert, and R. Kocaoğlu, "Machine learning approach equipped with neighbourhood component analysis for DDoS attack detection in software-defined networking," *Electronics*, vol. 10, no. 11, May 2021, doi: 10.3390/electronics10111227.

[2] S. Chakraborty, P. Kumar, and B. Sinha, "A study on DDoS attacks, danger and its prevention," *International Journal of Research and Analytical Reviews*, vol. 6, no. 2, pp. 10–15, 2019, doi: 10.1729/Journal.20847.

[3] J. Wang, L. Wang, and R. Wang, "A method of DDoS attack detection and mitigation for the comprehensive coordinated protection of SDN controllers," *Entropy*, vol. 25, no. 8, Aug. 2023, doi: 10.3390/e25081210.

[4] K. Salunke and U. Ragavendran, "Shield techniques for application layer DDoS attack in MANET: a methodological review," *Wireless Personal Communications*, vol. 120, no. 4, pp. 2773–2799, Oct. 2021, doi: 10.1007/s11277-021-08556-3.

[5] J. A. P. -Diaz, I. A. Valdovinos, K.-K. R. Choo, and D. Zhu, "A flexible SDN-based architecture for identifying and mitigating low-rate DDoS attacks using machine learning," *IEEE Access*, vol. 8, pp. 155859–155872, 2020, doi: 10.1109/ACCESS.2020.3019330.

[6] K. S. Sahoo *et al.*, "An evolutionary SVM model for DDOS attack detection in software defined networks," *IEEE Access*, vol. 8, pp. 132502–132513, 2020, doi: 10.1109/ACCESS.2020.3009733.

[7] S. Ullah, Z. Mahmood, N. Ali, T. Ahmad, and A. Buriro, "Machine learning-based dynamic attribute selection technique for DDoS attack classification IoT networks," *Computers*, vol. 12, no. 6, May 2023, doi: 10.3390/computers12060115.

[8] A. Chopra, S. Behal, and V. Sharma, "Evaluating machine learning algorithms to detect and classify DDoS attacks in IoT," in *2021 8th International Conference on Computing for Sustainable Global Development, INDIACom 2021*, 2021, pp. 517–521, doi: 10.1109/INDIACom51348.2021.00092.

[9] Y. I. Aljanabi, A. A. Majeed, K. H. Jihad, and B. A. Qader, "Detect and mitigate blockchain-based DDoS attacks using machine learning and smart contracts," *Informatica*, vol. 46, no. 7, Oct. 2022, doi: 10.31449/inf.v46i7.4033.

[10] L. Tan, Y. Pan, J. Wu, J. Zhou, H. Jiang, and Y. Deng, "A new framework for DDoS attack detection and defense in SDN environment," *IEEE Access*, vol. 8, pp. 161908–161919, 2020, doi: 10.1109/ACCESS.2020.3021435.

[11] U. Dudekula and P. Nalluri, "Linear fusion approach to convolutional neural networks for facial emotion recognition," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 25, no. 3, pp. 1489–1500, Mar. 2022, doi: 10.11591/ijeecs.v25.i3.pp1489-1500.

[12] E. D. L. Hoz, E. D. L. Hoz, A. Ortiz, J. Ortega, and B. Prieto, "PCA filtering and probabilistic SOM for network intrusion detection," *Neurocomputing*, vol. 164, pp. 71–81, Sep. 2015, doi: 10.1016/j.neucom.2014.09.083.

[13] Z. Xu *et al.*, "Software defect prediction based on kernel PCA and weighted extreme learning machine," *Information and Software Technology*, vol. 106, pp. 182–200, Feb. 2019, doi: 10.1016/j.infsof.2018.10.004.

[14] N. Ahuja, G. Singal, and D. Mukhopadhyay, "DDOS attack SDN dataset," *Mendeley Data, V1*, 2020, doi: 10.17632/jxpfjc64kr.1.

[15] D. Usen and N. Purnachand, "Qualitative analysis of emotion recognition systems using deep neural networks," in *Computer-Aided Developments: Electronics and Communication*, Boca Raton, Florida: CRC Press, 2019, pp-155–164, doi: 10.1201/9780429340710-19.

[16] N. Ahuja, G. Singal, D. Mukhopadhyay, and N. Kumar, "Automated DDOS attack detection in software defined networking," *Journal of Network and Computer Applications*, vol. 187, Aug. 2021, doi: 10.1016/j.jnca.2021.103108.

[17] R. A. Elsayed, R. A. Hamada, M. I. Abdalla, and S. A. Elsaid, "Securing IoT and SDN systems using deep-learning based automatic intrusion detection," *Ain Shams Engineering Journal*, vol. 14, no. 10, Oct. 2023, doi: 10.1016/j.asej.2023.102211.

[18] N. N. Tuan, P. H. Hung, N. D. Nghia, N. V. Tho, T. V. Phan, and N. H. Thanh, "A DDoS attack mitigation scheme in ISP networks using machine learning based on SDN," *Electronics*, vol. 9, no. 3, Feb. 2020, doi: 10.3390/electronics9030413.
[19] A. Manickam, J. Jiang, Y. Zhou, A. Sagar, R. Soundrapandiyan, and R. D. J. Samuel, "Automated pneumonia detection on chest X-ray images: a deep learning approach with different optimizers and transfer learning architectures," *Measurement*, vol. 184, Nov. 2021, doi: 10.1016/j.measurement.2021.109953.
[20] M. S. Elsayed, H. Z. Jahromi, M. M. Nazir, and A. D. Jurcut, "The role of CNN for intrusion detection systems: an improved CNN learning approach for SDNs," in *Future Access Enablers for Ubiquitous and Intelligent Infrastructures (FABULOUS 2021)*, Cham, Switzerland: Springer, 2021, pp. 91–104, doi: 10.1007/978-3-030-78459-1_7.
[21] N. S. Musa, N. M. Mirza, S. H. Rafique, A. M. Abdallah, and T. Murugan, "Machine learning and deep learning techniques for distributed denial of service anomaly detection in software defined networks—current research solutions," *IEEE Access*, vol. 12, pp. 17982–18011, 2024, doi: 10.1109/ACCESS.2024.3360868.
[22] H. Cheng, J. Liu, T. Xu, B. Ren, J. Mao, and W. Zhang, "Machine learning based low-rate DDoS attack detection for SDN enabled IoT networks," *International Journal of Sensor Networks*, vol. 34, no. 1, pp. 56–69, 2020, doi: 10.1504/IJSNET.2020.109720.
[23] S. Mehmood, R. Amin, J. Mustafa, M. Hussain, F. S. Alsubaei, and M. D. Zakaria, "Distributed denial of services (DDoS) attack detection in SDN using optimizer-equipped CNN-MLP" *PLOS ONE*, vol. 20, no. 1, 2025, doi: 10.1371/journal.pone.0312425.
[24] V. Blanco, A. Japón, and J. Puerto, "Multiclass optimal classification trees with SVM-splits," *Machine Learning*, vol. 112, no. 12, pp. 4905–4928, Dec. 2023, doi: 10.1007/s10994-023-06366-1.
[25] M. S. Elsayed, N.-A. L. -Khac, S. Dev, and A. D. Jurcut, "Network anomaly detection using LSTM based autoencoder," in *16th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, Nov. 2020, pp. 37–45, doi: 10.1145/3416013.3426457.

BIOGRAPHIES OF AUTHORS



Bandaru Venkata Chandrasekhar is research scholar at Department of Computer Science, Bharat Institute of Higher Education and Research, Selaiyur, Chennai, India. He received his master degree in Computer Science and Engineering from Jawaharlal Nehru Technological University Anantapur, Andhra Pradesh. His main research areas include software define networking, machine learning, and deep learning. He can be contacted at email: bvchandraphd@gmail.com.



Dr. Rajabhushanam Cherukuri is working as professor at Department of Computer Science, Bharat institute of Higher Education and Research, Selaiyur, Chennai. He did his Ph.D. in Computer Science from Louisiana State University. Possess expertise in computer science engineering, GIS/geospatial management, application development and analysis, spatial database, and enterprise GIS design and implementation. A total of 20 years of strong project management experience in Fortune 500 companies and was responsible for leading team sizes of 50-100 personnel. Five years of senior management in the roles of head of geospatial division, business development manager, and program/project manager. Expert in relational database management systems; object-oriented and SQL based application development. He can be contacted at email: rajabhushanamc.cse@bharathuniv.ac.in.